```
from google.colab import drive
drive.mount('/content/gdrive')
%cd ./gdrive/'My Drive'/"videoexp"

     Drive already mounted at /content/gdrive; to attempt to forcibly remount, call drive.mount("/
     [Errno 2] No such file or directory: './gdrive/My Drive/videoexp'
     /content/gdrive/My Drive/videoexp
```

```
import os
import numpy as np
from PIL import Image
from copy import deepcopy
import matplotlib.pyplot as plt
import keras
from keras.models import Sequential
from keras.layers import Dense, Conv2D, BatchNormalization, Conv2DTranspose, Activation, Flatten, D
from keras.layers.advanced_activations import LeakyReLU
from keras.preprocessing.image import ImageDataGenerator
from keras.models import load_model
from keras.callbacks import EarlyStopping
```

```
#パラメータ
#======================================
#教師画像の親フォルダ
f = 'src/'


#ミニバッチサイズ（教師データ数の公約数にしてください）
batch_size = 5

#乱数列の次元
z_dim = 100

#discriminatorの学習率
opt = keras.optimizers.Adam(lr=0.0002)

#画像を保存するフォルダ
img_f = 'AE_img/'
#重みを保存するフォルダ
para_f = 'AE_para/'
#======================================


#教師データ読み込み
#======================================
x_train = []
originalx_train = []
files = os.listdir(f)
files = os.listdir(f)
for file in files:
    img = Image.open(f + file).convert("RGB"); img.close
    originalimage = img
    originalx_train.append(np.array(originalimage))
    img_resize = img.resize((28, 28))
    x_train.append(np.array(img_resize))

x_train = np.array(x_train)

#-1～+1に規格化
x_train = (x_train - 127.5) / 127.5



print(x_train)
```

```
[[[[-0.09019608 -0.03529412 -0.69411765]
   [-0.27058824 -0.23921569 -0.73333333]
   [-0.37254902 -0.34117647 -0.74901961]
   ...
   [ 0.65490196  0.78823529  0.98431373]
   [ 0.67058824  0.78823529  0.99215686]
   [ 0.64705882  0.77254902  0.98431373]]

  [[-0.12941176 -0.05098039 -0.69411765]
   [-0.17647059 -0.11372549 -0.71764706]
   [-0.30196078 -0.25490196 -0.75686275]
   ...
   [ 0.68627451  0.79607843  0.99215686]
   [ 0.69411765  0.79607843  1.        ]
   [ 0.67843137  0.78039216  0.98431373]]

  [[-0.12941176 -0.14509804 -0.56862745]
   [-0.12156863 -0.06666667 -0.61568627]
   [-0.24705882 -0.23921569 -0.69411765]
   ...
   [ 0.7254902   0.81960784  0.99215686]
   [ 0.71764706  0.79607843  0.98431373]
   [ 0.69411765  0.78039216  0.99215686]]

  ...

  [[-0.12941176 -0.01960784 -0.68627451]
   [-0.12156863 -0.01960784 -0.7254902 ]
   [-0.04313725  0.12941176 -0.65490196]
   ...
   [ 0.34117647  0.49803922 -0.54509804]
   [ 0.41176471  0.59215686 -0.51372549]
   [ 0.36470588  0.54509804 -0.52941176]]

  [[-0.1372549  -0.09803922 -0.67058824]
   [ 0.10588235  0.16862745 -0.65490196]
   [-0.01176471  0.0745098  -0.6627451 ]
   ...
   [ 0.17647059  0.30196078 -0.61568627]
   [ 0.38039216  0.5372549  -0.5372549 ]
   [ 0.31764706  0.49019608 -0.52941176]]

  [[ 0.04313725  0.04313725 -0.6        ]
```

```
   [ 0.24705882  0.37254902 -0.62352941]
   [ 0.30196078  0.42745098 -0.58431373]
   ...
   [ 0.34117647  0.45098039 -0.50588235]
   [ 0.34117647  0.48235294 -0.5372549 ]
   [ 0.28627451  0.45098039 -0.52941176]]]


 [[[ 0.06666667  0.09019608 -0.67058824]
   [-0.14509804 -0.1372549  -0.70196078]
   [-0.29411765 -0.33333333 -0.65490196]
   ...
   [ 0.63137255  0.79607843  0.98431373]
   [ 0.63137255  0.78823529  0.98431373]
   [ 0.59215686  0.76470588  0.98431373]]
```

```
x_train.shape
```

```
(30, 28, 28, 3)
```

```
vectorxtrain = x_train.reshape(30,28*28*3)
```

```
vectorxtrain[0]
```

```
array([-0.09019608, -0.03529412, -0.69411765, ...,  0.28627451,
        0.45098039, -0.52941176])
```

```
vectorxtrain[0].shape
```

```
(2352,)
```

```
vectorxtrain.shape
```

```
(30, 2352)
```

```
import numpy as np
from sklearn.decomposition import PCA
import matplotlib.pyplot as plt

pca = PCA()
```

```
n_comp = 20

pca = PCA(n_components=n_comp)

pca.fit(vectorxtrain)
#20次元の潜在空間に変換して圧縮
X_train_latent = pca.transform(vectorxtrain)
```

```
X_train_latent.shape
```

```
    (30, 20)
```

```
X_train_latent[0]
```

```
    array([-17.29425541,  -7.91048974,  -5.56915524,   0.54166923,
             1.34412518,  -0.92439218,  -2.7131581 ,   0.32917079,
             7.20120374,  -3.92984657,   6.21180207,  -2.69152913,
             0.85772159,   0.26090921,  -0.36896734,   1.87368534,
             0.15607904,  -2.27240563,  -2.05087192,   0.50682253])
```

```
import numpy as np
import matplotlib.pyplot as plt
import matplotlib.font_manager
from sklearn import svm

# fit the model
clf = svm.OneClassSVM(nu=0.1, kernel='rbf', gamma=0.03)
clf.fit(X_train_latent)
```

```
    OneClassSVM(cache_size=200, coef0=0.0, degree=3, gamma=0.03, kernel='rbf',
                max_iter=-1, nu=0.1, shrinking=True, tol=0.001, verbose=False)
```

```
y_pred_train_latent = clf.predict(X_train_latent)
```

```
print(y_pred_train_latent)
```

```
    [-1  1 -1 -1  1 -1 -1 -1 -1  1  1  1  1  1 -1  1 -1 -1  1  1  1 -1 -1 -1
     -1 -1 -1 -1  1 -1]
```

```
y_pred_train_latent.shape
```

```
    (30,)
```

```
kara = []
for i in range(30):
   if y_pred_train_latent[i]==-1:
              kara.append(i)
kara
```

```
    [0, 2, 3, 5, 6, 7, 8, 14, 16, 17, 21, 22, 23, 24, 25, 26, 27, 29]
```

```
len(kara)
```

```
    18
```

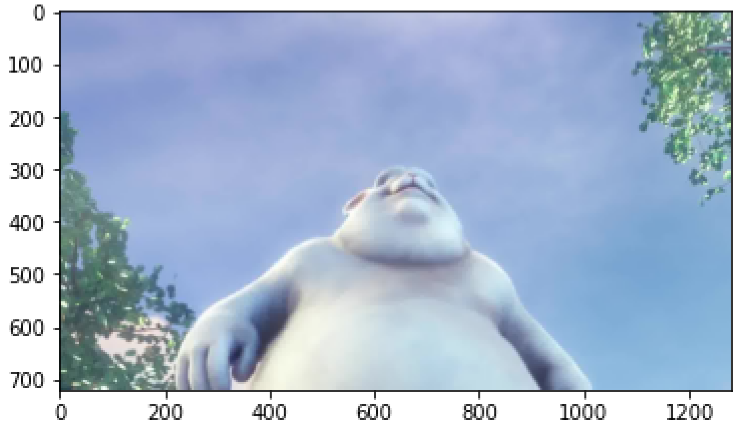
```
originalx_train[0].shape
```

```
    (720, 1280, 3)
```

```
import matplotlib.pyplot as plt
import matplotlib.font_manager
```

```
plt.imshow(originalx_train[10], cmap=plt.cm.gray_r)

     <matplotlib.image.AxesImage at 0x7efbf9d58b10>
```



```
from google.colab import drive
drive.mount('/content/gdrive')

     Drive already mounted at /content/gdrive; to attempt to forcibly remount, call drive.mount("/
```

```
%cd ./gdrive/'My Drive'/"videoexp"

     [Errno 2] No such file or directory: './gdrive/My Drive/videoexp'
     /content/gdrive/My Drive/videoexp
```

```
import numpy as np
import cv2
```

```
for i in range(len(kara)):
  cv2.imwrite(f'comb{i:03d}.jpg', cv2.cvtColor(originalx_train[kara[i]]                        , cv2
```

```
import glob

from PIL import Image

frames = []
images = sorted(glob.glob("./*.jpg"))

print(images)

     ['./comb000.jpg', './comb001.jpg', './comb002.jpg', './comb003.jpg', './comb004.jpg', './comb
```

```
for image in images:
    new_frame = Image.open(image)
    frames.append(new_frame)
```

```
frames[0].save('jpg_to_gif.gif',
               format='GIF',
               append_images=frames[1:],
               save_all=True,
               duration=500,
               loop=0)
```